Permission to Speak: A Novel Formal Foundation for Access Control

Oleg Sokolsky

Nikhil Dinesh, Insup Lee, Aravind Joshi

Report Documentation Page

Form Approved
OMB No. 0704-0188

1. REPORT DATE **04 NOV 2009**	2. REPORT TYPE	3. DATES COVERED **00-00-2009 to 00-00-2009**

4. TITLE AND SUBTITLE **Permission to Speak: A Novel Formal Foundation for Access Control**	5a. CONTRACT NUMBER
	5b. GRANT NUMBER
	5c. PROGRAM ELEMENT NUMBER
6. AUTHOR(S)	5d. PROJECT NUMBER
	5e. TASK NUMBER
	5f. WORK UNIT NUMBER
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) **University of Pennsylvania,Computer and Information Science,Philadelphia,PA,19104**	8. PERFORMING ORGANIZATION REPORT NUMBER
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	10. SPONSOR/MONITOR’S ACRONYM(S)
	11. SPONSOR/MONITOR’S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution unlimited

13. SUPPLEMENTARY NOTES
ONR MURI Review, Nov 2009.

14. ABSTRACT

15. SUBJECT TERMS

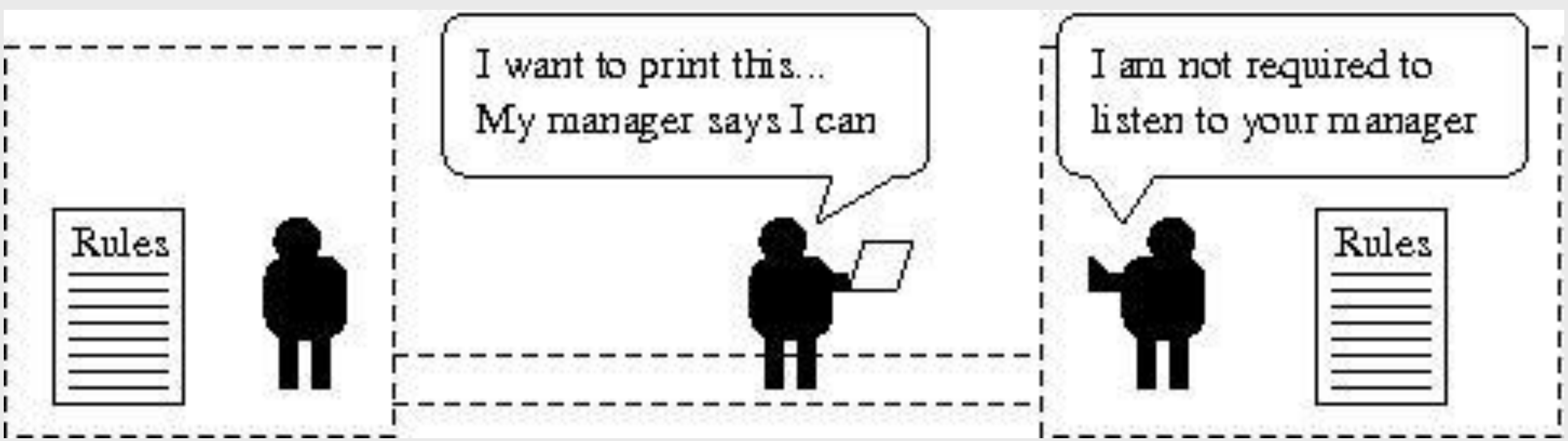
16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT **unclassified**	b. ABSTRACT **unclassified**	c. THIS PAGE **unclassified**	**Same as Report (SAR)**	**9**	

Outline

- Motivation
  - Distributed, multi-authority access control
  - Compliance checking and blame assignment
- Formal representation
  - Delegation and obligation
  - Permission as provability
- Access control and conformance checking
  - System architecture
- Summary

Motivation and problem statement

- Main problem of access control:
  - Should a request for service be granted?
- In a distributed system with multiple authorities:
  - Which policies need to be consulted?
  - Which policies are violated and who is to blame?



Delegation and obligation

- "saying" is a common operator in access control logics
  - Captures both policy and credential introduction
  - Policies are typically obligations and credentials are typically permissions
  - Obligations and permissions are often implicit and must be deduced by the checker
- Explicit permissions and obligations
  - Deontic operators $\boldsymbol{P}_A \phi$, $\boldsymbol{O}_A \phi$

L_{PS}:logic and policies

- L_{PS} is a decidable logic with complete semantics
- Key formal device: axiom of representation

$$\left(says_{l(A)} \left(P_B \, says_{l(B)} \varphi \right) \wedge says_{l(B)} \varphi \right) \Rightarrow says_{l(A)} \varphi$$

- A policy is a collection of sequents

$$(id)\varphi \mapsto \psi$$

  - True preconditions must have true postconditions
  - Postconditions make more preconditions true

Contributions to science

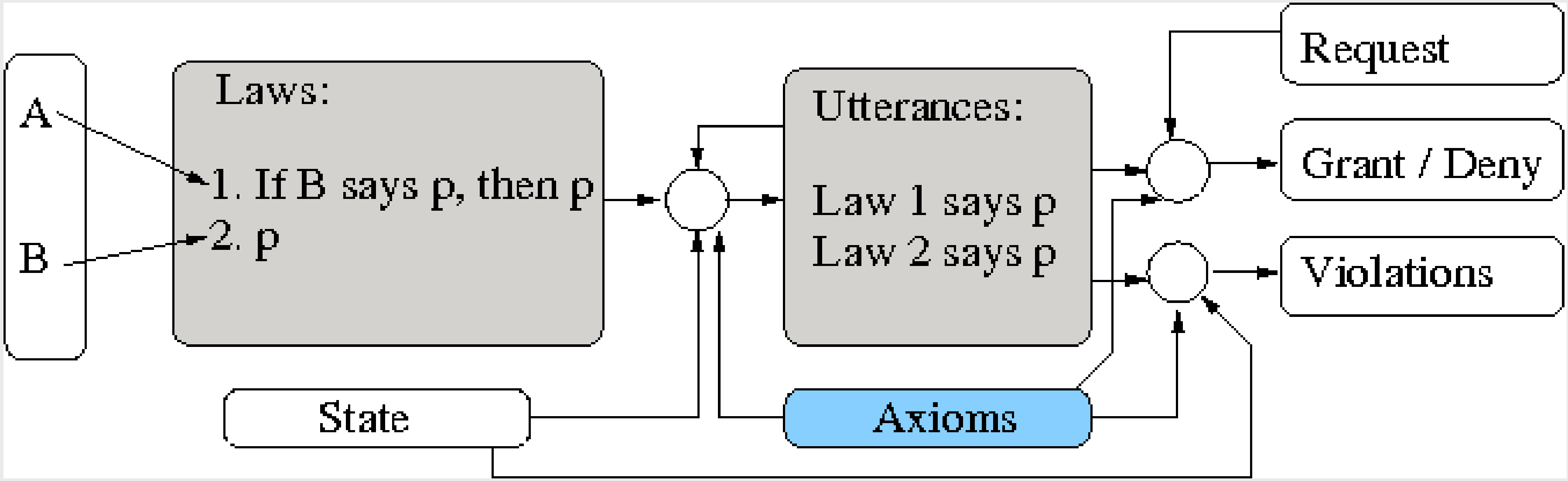
- Uniform treatment of access control and conformance
  - Access control is verification of permissions
  - Conformance is satisfaction of obligations
  - Both are formalized as provability of statements in the logic
- Clarified semantics of deontic modalities
  - Nested permissions and obligations
  - Positive and negative permissions

Nested deontic modalities

- Parents (A) should not let their children (B) play by the road
  - Multiple possible interpretations:
    - A should not give B permission to play (positive permission)
    - A should tell B not to play (negative permission)
    - A should physically prevent B from playing
  - Each interpretation make sense in some context
- Alternation with saying solves the problem
  - “require to allow” becomes “require to make a rule…”
    - $O_A\left(\neg says_{l(A)} P_B\, play_{road}(B)\right)$
    - $O_A\left(says_{l(A)} O_B \neg play_{road}(B)\right)$

System architecture

- Principals introduce laws
- Logic programming engine computes *utterances*, ground saying terms
- Request is granted if utterances contain a permission for it



Future work: quantitative evaluation

- L_{PS} can be used as an alternative to Keynote in the QuanTM architecture
- A tighter integration with the reputation manager will be more efficient
- Quantitative semantics for L_{PS} will combine TDG construction and evaluation
  - Supported by the logic programming framework of L_{PS}
  - Similar to probabilistic Datalog semantics